**Create project and provide permissions:**

When Tableau Desktop users publish a workbook or data source to a site on Tableau Server, they can select a project to publish it to. If they don’t specify a project, their content is published to the **Default** project.

As an administrator, you can create projects to hold and organize related content, or to delegate content management. Content on Tableau Server are workbooks, views, and data sources, and the projects that hold them. If the Data Management Add-on is present, content types also include Flows and Data Roles.

**Why use projects**

Projects help you to create a scalable process for managing access to the content published to Tableau Server. Advantages they have include:

* They enable administrators to delegate content management to project leaders who work with the content more closely, without having to give them administrator access to site or server settings.
  + Project leaders can create nested projects under their top-level project, enabling them to maintain their team’s content within a single hierarchy.
  + **Note:** Project owners can delete top-level projects they own. Project leaders cannot delete top-level projects.
* They can make the site easier to navigate for self-service users.
  + They segment the Tableau Server site into areas that give users access based on how they use the data published to those areas, or on the Tableau user group they work with.
  + You can hide projects from groups who don’t need to use them, create a distinguishable project-naming scheme, and take advantage of project descriptions to clarify how to use the project.
* They enable you to track permissions effectively.
  + You can create groups based on the level of content access users in the group need, and set default permissions on projects. This enables you to know exactly which capabilities new users get by default, and likewise which capabilities all users get when a new project is created.

### **how to create project hierarchies (example)**

Many organizations have several or more distinct groups of Tableau users, each with its own priorities and leaders. These groups might share some organization-wide content (or even draw from an org-wide pool of data sources), but primarily they use data and reports that are specific to their team. In this or similar scenario, an example for using project hierarchies might look as follows:

1. You, as a site or server administrator, can create top-level projects for each of your distinct Tableau teams.
2. On each top-level project, you assign the Project Leader status to team leads, and change project ownership. Project leaders effectively are the content administrators, so it’s important that they understand how permissions work in Tableau, along with Tableau content management best practices.
3. Each project leader can manage their project, creating the structure within the project that works for their team. That is, they can create child projects they need, based on how their team members collaborate and share data and reports.

### **Why not use sites?**

If you manage your own Tableau Server deployment, you can create as many sites as you want. However, for managing data and reports across your company, projects allow the flexibility you need to administer shared data and reports, and users who might belong to multiple groups. Many Tableau administrators configure projects as described in the previous section, to expose only what’s necessary to users who need to work with it. Projects work better than sites for evolving content from development to staging to production.

Sites work well when content can remain completely separate during all phases, and there is little to no user overlap. A good (and common) example for using multiple sites is to create a site for each of multiple external clients, whose published content you manage as a consultant or vendor. Our own Tableau Online is an example of this on a large scale. Another example might be to use a separate site for sensitive content that you want only specific Human Resources or medical staff to use.

## Project-level administration

As a server or site administrator, you can delegate administration of projects and their content, without exposing access to your site or server settings. You can do this by changing the ownership of a project or granting a group or user project leader status on a project.

The **project owner** is always one individual user. By default, the user who creates a project is its owner. The project owner has administrative access to the project and content in it—including making someone else the owner and assigning Project Leader permissions.

The **project leader** setting provides a way to allow multiple users administrative access to a project, its child projects, and all workbooks and data sources in those projects.

A project leader does not have to be a project owner or administrator. In addition to server and site administrators, the full scope of Project Leader permissions is available to users with a **Creator** or **Explorer (can publish)** site role.

### **Actions project-level administrators can take on projects**

Project leaders and owners can perform the tasks in the following list, as can server or site administrators.

* **Create and delete projects as follows:**
  + Server or site administrators can create or delete top-level or nested projects anywhere on the site.
  + Project owners and project leaders can create and delete child (nested) projects in projects they own, or on which they have project leader status.
* Project owners can change ownership of their projects. Both project owners and project leaders can assign the project leader status to groups or users.
  + As a project leader or owner, if you assign the project leader status to someone else, remember that full access to this role’s capabilities depends on the user’s site role, as specified earlier in this section.
* Set permissions for a project, as well as the child projects, workbooks, and data sources in it.
* Lock permissions to apply the project’s default settings to all workbooks, data sources, and optionally child projects and their content. For information, see [Lock content permissions](https://help.tableau.com/current/server/en-us/permissions.htm#LockProject).
  + Permissions can be modified only from the project they’re locked on. Only admins, the owner, or project leaders can change permissions.
* Move workbooks and data sources to another project to which they have project leader or owner access. Moving the project can affect permissions. For information, see [Move content](https://help.tableau.com/current/server/en-us/permissions.htm#MoveContent).
* Run, add, or remove extract refresh schedules.

## Prevent publishers from changing permissions on content they own

Administrators and project leaders can prevent users from changing the permissions for workbooks and data sources in a project hierarchy. For example, you can disable the option to set permissions during the publishing process, and prevent publishers and content owners from changing them after publishing. To do this, you lock content permissions to the project.

# **Permissions**

Permissions determine how users can interact with content such as workbooks and data sources. Permissions are set in the permission dialog or via the [REST APILink opens in a new window](https://help.tableau.com/current/api/rest_api/en-us/REST/rest_api_ref_permissions.htm). At the top of the dialog, permission rules configure capabilities for groups or users. Below, the permissions grid displays the effective permissions for users.

Permissions fundamentals

**Projects and groups**

Tableau sites use projects to organize content and groups to organize users. Managing permissions is easier when permission rules are:

* Set at the project level instead of on individual pieces of content.
* Established for groups instead of individuals.

Permissions can only be established for users, groups, projects, or content that already exist. For more information about creating users and groups, creating projects, and publishing content, see [Manage Users and Groups](https://help.tableau.com/current/server/en-us/users.htm), [Use Projects to Manage Content Access](https://help.tableau.com/current/server/en-us/projects.htm) , and [Publish Data Sources and WorkbooksLink opens in a new window](https://help.tableau.com/current/pro/desktop/en-us/publish_overview.htm).

**Capabilities and permission rules**

Permissions are made up of capabilities—the ability to perform actions like view content, web edit, download data sources, or delete content. Permission rules establish what capabilities are allowed or denied for a user or group on a piece of content.

To set permissions at the project level:

1. Navigate to the project
2. Open the Actions menu (...) and click **Permissions**. The permissions dialog opens.

This dialog has two main areas: permission rules at the top and the effective permissions grid below. Each content type (Project, Workbooks, Data Sources, Flows, Data Roles) has a tab. The image below shows the Workbook tab.

With a row selected at the top, the effective permissions grid populates. Use this to verify permissions. Hovering provides information about why the capability is allowed or denied for that specific user.

1. To modify an existing permission rule, select the appropriate tab for that content type and click a capability.
2. To create a new rule, click **+ Add Group/User Rule** and start typing to search for a group or user. For each tab, choose an existing template from the drop-down box or create a custom rule by clicking the capabilities.

One click sets the capability to **Allowed**, two clicks sets it to **Denied**, and a third click clears the selection (**Unspecified**).

1. When finished, click **Save**.

### **Templates**

Templates group sets of capabilities that are often assigned together based on common user scenarios, **View**, **Explore**, **Publish**, and **Administer**. Assigning a template sets its included capabilities to **Allowed**, with the rest left as **Unspecified**. The templates are cumulative, so the Explore template includes everything from the View template plus additional capabilities. All content also has a template for **None** (which sets all capabilities to unspecified) and **Denied** (which sets all capabilities to denied).

Templates are meant to be a starting point and can be adjusted after they are applied. Capabilities can also be granted or denied without using a template at all. In both cases, the template column will then show **Custom**.

### **Copy and paste permissions**

If there is a permission rule that needs to be assigned to multiple groups or users, you can copy and paste from one rule to another. You can’t copy from or paste onto a rule that involves Project Leader status.

1. Open the action menu (...) for the existing rule you want to copy from and select **Copy Permissions**. This will only be available when the rule is not in edit mode.
2. Select an existing rule you want to paste over. You can also create a new rule by clicking **+ Add Group/User Rule** and selecting a group or user.
3. Open the action menu (...) and select **Paste Permissions**.

#### **View template**

**View** allows a user to see the project. If a user hasn’t been granted the view capability, the project won’t be visible to them. Granting the view capability for a project does not mean a user can see any content in the project, just the existence of the project itself.

#### **Publish template**

**Publish** allows a user to publish content to the project from Tableau Desktop or Tableau Prep Builder. The publish capability is also required to move content into the project or save content to the project from web authoring. Prior to 2020.1, this capability was called Save.

**Deleting**: Content can only exist inside a project. Only administrators can create and delete top-level projects, but project leaders can create or delete nested projects. Deleting projects also deletes all the content and nested projects they contain. To delete a project without losing its content, move the content to another project first. Deleting projects can’t be undone.

**Ownership**: A project can have multiple project leaders, but each project has exactly one owner. By default, a project is owned by the user who created it. A project’s owner can be changed (by the existing owner or an administrator, but not a project leader) to any user with a site role of Explorer (can publish) or Creator, or an administrator site role. Project ownership can be changed regardless of whether the project permissions are locked. Note that this refers to project ownership. Content ownership can be changed by project owners, project leaders, and administrators.

### **Lock content permissions**

Permission rules set at the project level act as a default for content saved in that project and any nested projects it contains. Whether those project-level default rules are enforced or only preliminary depends on the **content permission** setting. This setting can be configured in two ways, either **Locked** (recommended) or **Customizable**. Locking a project removes the ability for content owners to modify the permission rules on their content. Locking permissions can be applied to nested projects or just to the parent project itself.

* When the content permissions are **locked (including nested projects)**, permission rules set at the project level are enforced for all content in the project and all nested projects. (This was the default behavior for locking projects prior to 2020.1)
* When the content permissions are **locked** (not including nested projects), permission rules set at the project level are enforced for content in the project, but nested projects can be configured independently with their own permission rules and as locked or customizable. (This is new behavior for locking projects as of 2020.1)
* When the content permissions are **customizable**, permission rules set at the project level are applied to all content in the project by default. However, permission rules can be modified for individual pieces of content during or after publishing. (This was called **Managed by the owner** prior to 2020.1)

**Note**: Whether permission rules are locked or customizable, the permissions on content are always applied. Locked and customizable refer only to how project-level permissions are inherited by content in the project and who can change them. Even in a project with customizable permissions, only specific users can modify permissions (content or project owner, project leader, admins, or those with the Set Permission capability).

In a locked project:

* The project permission rules per content type are applied to all content.
* Only administrators, project owners, and project leaders can modify permissions.
* Content owners lose the Set Permission capability but retain all other capabilities on their content.
* Permissions are predictable for all content in the project.

In a customizable project:

* The project permission rules are applied by default when content is published into the project or nested projects are created, but permissions can be modified during publication or after the content is created.
* Any user with the Set Permissions capability can modify permission rules for that content.
* Content owners have all capabilities on their content.
* Permissions can be different across content in the project.

#### **Move projects and content**

When a project is moved into another project, the permissions settings on the project being moved are maintained unless the destination project is scoped to include nested projects.

* If the destination project is set to **locked (including nested projects)**, the permissions for the project being moved are overwritten.
* If the destination project is set to **locked** (not including nested projects), the permissions for the project being moved are not overwritten. Whether or not the moved project is locked or customizable is preserved from its original setting.
* If the destination project is set to **customizable**, the permissions for the project being moved are not overwritten but they are now editable.
  + If the project being moved was previously nested under a parent that was locked (including nested projects), when moved, the project takes on the setting of locked (including nested projects) and becomes the managing project for any projects it contains. Note: This is the same outcome if a project is moved to become a top-level project.

## Permission settings for specific scenarios

Certain actions require combinations of permission capabilities and possibly site roles. The following are some common scenarios and their necessary permission configurations

### **Saving, publishing, and overwriting**

In the context of permissions, saving is essentially publishing. As such, the **Overwrite** and **Save a Copy** capabilities can only be given to users with a site role that allows publishing: Administrator, Creator, or Explorer (can publish). Explorer or Viewer site roles can’t publish, overwrite, or save a copy.

(Prior to version 2020.1, the Publish and Overwrite capabilities were called Save, and the Download Workbook/Save a Copy capability was called Download Workbook/Save As.)

* The **Publish** capability for a project allows a user to publish content into that project.
* The **Overwrite** capability allows a user to save over an existing piece of content; they become the owner.
* The **Save a Copy** capability allows a user to save a new copy of the content. This is usually done in conjunction with web authoring and means the user can save their modifications.

It’s important to note that users aren’t able to Save or Save As a piece of content unless they have the **Publish** capability for at least one project, because all content must be published into a project. Without the **Publish** capability at the project level, the content can’t be published.

### **Data access for published Tableau data sources**

Data sources published to a Tableau site can have native authentication as well as permissions within the Tableau environment.

When the data source is published to the Tableau site, the publisher can choose how to [Set Credentials for Accessing Your Published DataLink opens in a new window](https://help.tableau.com/current/pro/desktop/en-gb/publishing_sharing_authentication.htm#workbook-connections-to-tableau-data-sources) which addresses how data source credentials are handled (such as requiring users to log into a database or enter their credentials for Google Sheets). This authentication is controlled by whatever technology holds the data. This can be embedded when the data source is published, or the data source publisher can choose to prompt the user for their credentials to the data source. For more information, see [Publish a Data SourceLink opens in a new window](https://help.tableau.com/current/pro/desktop/en-us/publish_datasources.htm#publishing-steps).

There are also data source capabilities that allow or deny users the ability to see (**View**) and connect to the published data source (**Connect**) in the context of Tableau. These capabilities are set like any other permissions in Tableau.

When a workbook is published that uses a published data source, the author can control how the Tableau authentication will behave for someone consuming the workbook. The author sets the workbook’s access to the published data source, either as **Embed password** (using the author’s Connect access to the data source) or **Prompt users** (using the Connect access of the person viewing the workbook), which may require data source authentication as well.

* When the workbook is set to **Embed password**, anyone who looks at the workbook will see the data based on the author’s access to the data source.
* If the workbook is set to **Prompt users**, the Tableau-controlled access is checked for the data source. The person consuming the workbook must have the Connect capability for the published data source to see the data. If the published data source is also set to Prompt user, the viewer must also enter their credentials for the data source itself.

### **Move content**

To move an item, open its Action menu (...) and click **Move**. Select the new project for the item, then click **Move Content**. If Move is unavailable or there are no available destination projects, verify the appropriate conditions are met:

* Administrators can always move content and projects to any location.
* Project leaders and project owners can move content and nested projects among their projects.
  + Note that non-administrators can’t move projects to become top-level projects
* Other users can move content only if all three of the following requirements are met:
  + Creator or Explorer (Can Publish) site role.
  + Publishing rights (**View** and **Publish** capabilities) for the destination project
  + Owner of the content, or—for workbooks and flows—having the **Move** capability.

When a project is moved, the permissions for its content might change.

* Project leaders or project owners always gain permissions for items moved into their projects.
* When a project is moved into a **locked (including nested)** project, the permission templates for the locked project are enforced on the moved project and all its content and nested projects. (Note that this might strip the user moving the project of their ability to move it again if they don’t have the correct permissions in the locked project.)
* When a project is moved into an **unlocked** project (customizable), the existing permissions are retained for the moved project and its content. If the project leader status has only implicitly been granted (from a higher-level project), that status is removed, though any explicitly set project leader status is retained.

**User Roles in Tableau Server**

There are 8 different site roles that can be assigned to your Tableau Server users. The roles determine how much access each user has to the content stored on the server and are assigned at a ‘Site’ level. Individual users can therefore have different levels of access depending on what section of the server they are in (see Anna’s blog for a bit more info on [Sites](http://www.thedataschool.co.uk/anna-noble/tableau-server-user-interface-diagram-protect-data/)). Roles can be assigned when the user is created or they can be edited for users that have already been created.

* **Server Administrator**: Full access to all server and site functionality, all content on the server, and all users. They can basically do whatever they want.
* **Site Administrator**: Like a Server Admin but only for individual sites, not the server as a whole. They have unrestricted access to the only the site that they are Admin for. Can add users and change site roles on their site but only if the Server Admin has given them permission. A user can be a site administrator on multiple sites
* **Publisher**: Publishers can connect to the server from Tableau desktop and upload workbooks and data sources. They can also interact with and download the views that are on the  server.
* **Interactor**: Interactors can browse the server, and interact with the published views but they are not allowed to publish.
* **Viewer:**As the name suggests, they can view workbooks, however they will only get a flat image and cannot interact with it.
* **Unlicensed**: Unlicensed users cannot sign in to the server. Mainly used to free up license spots without removing the user (which would make the content they have created unreachable).
* There are also **Viewer (can publish) and Unlicensed (can publish)**roles that would primarily be used for automated scripting activities.

User roles are not the only way of controlling who can see and do what on your Tableau server. Permissions can also be set on Projects, Workbooks and Data Sources to control user privileges. It is important to be aware that the site role of a user will overrule any permissions that may be set at a project level so it is important to get the roles right when the users are set up. For example, if you set an employee’s workbook permissions to ‘Editor’ but their role is only Interactor then they will only be able to view and interact with workbooks, not create or edit them.

**Extract Fail**

For a variety of reasons, the data extract refresh may fail to work as scheduled.

I summarized all the failures as 5 types.

**- Bad data file path**

When the data is in a flat file, such as Excel, Access, CSV etc, the server can't find the data file. One of the most likely cases is the file path is in local format, such as C:\folder\data.csv. The server actually expects an UNC path such as \\machine\folder\data.csv

**- DB connect fail**

The connection to the database failed. Need to fix the connection settings or the network etc.

**- DB query fail**

The query worked when being published. However the database tables may have changed. The query can fail in this case. The solution is to revise the query.

**- Data engine fail**

The service may be down. Need a reboot.

**- Suspended**

There are too many failures. Need to download the workbook or data source and republish it. The failure will resurface and the error message will tell you what the failure is.